

FIG. 1A

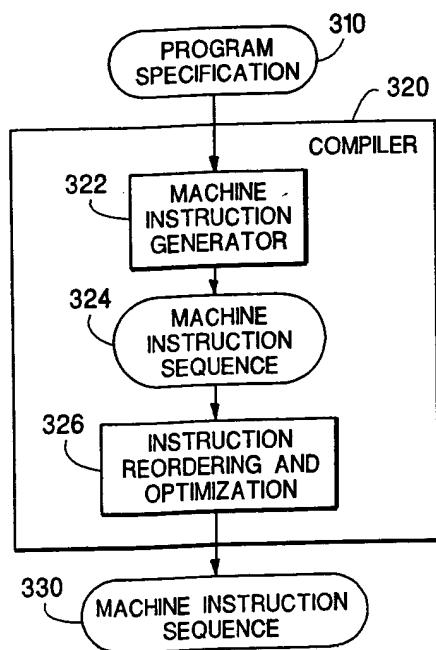


FIG. 3A

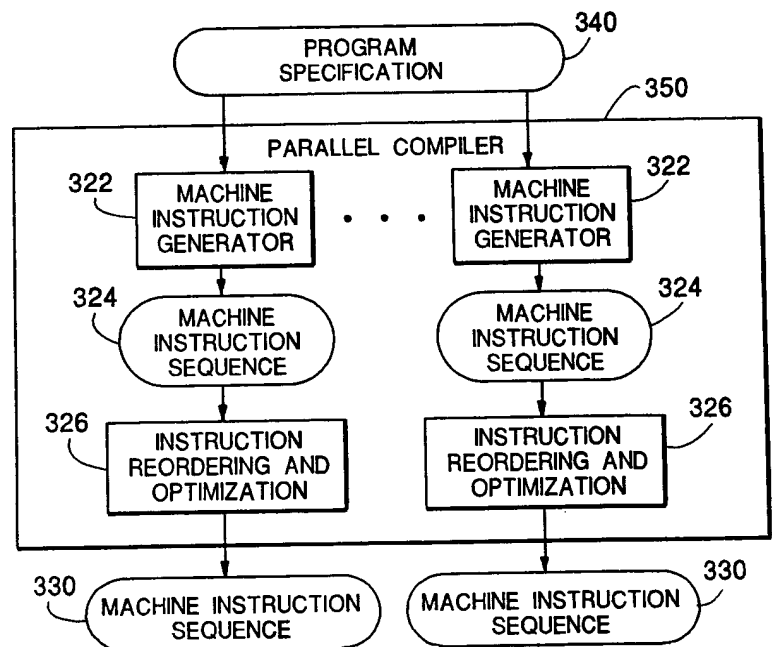


FIG. 3B

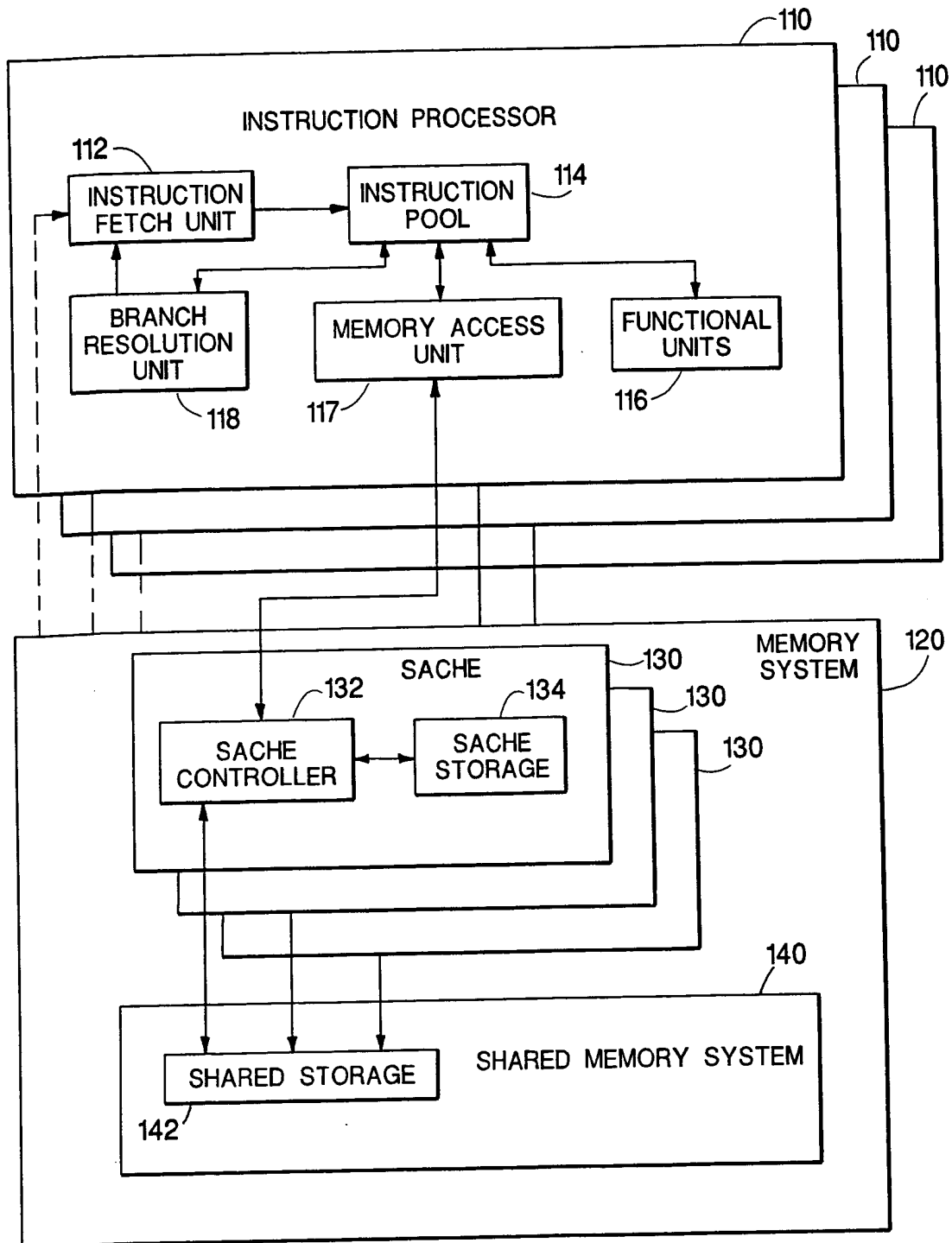
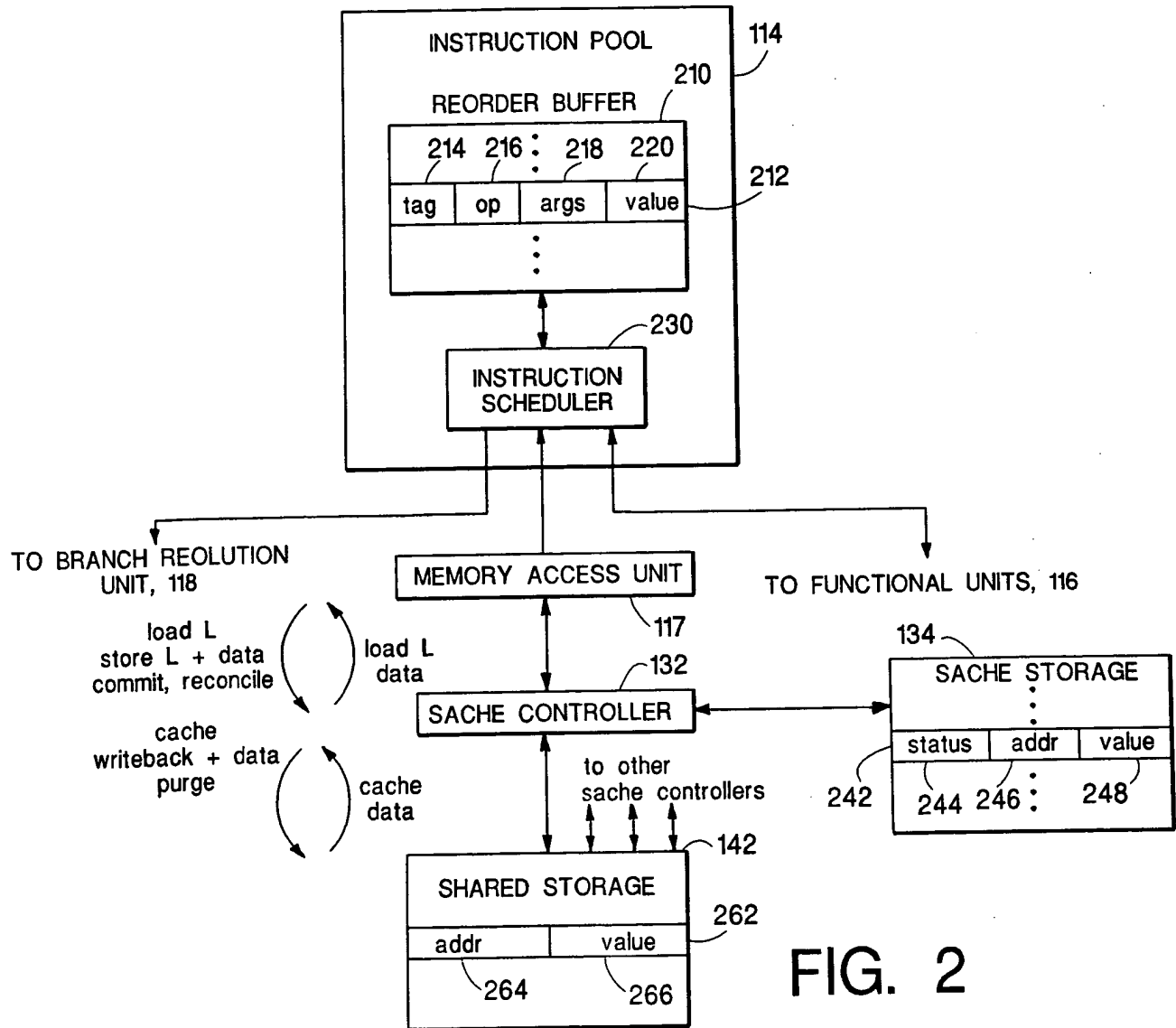


FIG. 1B



Process Load(addr): 410

```
411  IF Cell(addr,-,Invalid) THEN
412      CREATE Cell(addr,-,Cache-Pending)
413      SEND Cache-Request(addr) message
414      WAIT FOR Cache(addr,val) return message
415      SET Cell(addr,val,Clean)
416      RETURN val
417  ELSE IF Cell(addr,val,-)
418      RETURN val
419  ENDIF
```

FIG. 4A

Process Reconcile(addr) message: 430

```
431  IF Cell(addr,-,Clean) THEN
432      DELETE addr from sache storage (Cell(addr,-,Invalid))
433  END IF
434  RETURN ack
```

FIG. 4B

Process StoreL(addr,val) message: **460**

```
461 IF Cell(addr,-,Invalid) THEN
462     CREATE Cell(addr,-,-)
463 END IF
464 SET Cell(addr,val,Dirty)
465 RETURN ack
```

FIG. 4C

Process Commit(addr) message: **470**

```
471 IF Cell(addr,val,Dirty) THEN
472     SET Cell(addr,val,Writeback-Pending)
473     SEND Writeback(addr,val) message
474     WAIT FOR ack message
475     SET Cell(addr,val,Clean)
476 END IF
477 RETURN acknowledgment
```

FIG. 4D

Create Cell(addr,-,status): **480**

```
481 IF no space available in sache storage THEN
482     SELECT Cell(addr',val',status') such that status' = Clean or Dirty
483     IF status' = Dirty THEN
484         SEND Writeback(addr',val') message
485         WAIT FOR ack message
486     END IF
487     DELETE Cell(addr',val',status')
488 END IF
489 SET Cell(addr,-,status) in available cell
```

FIG. 4E

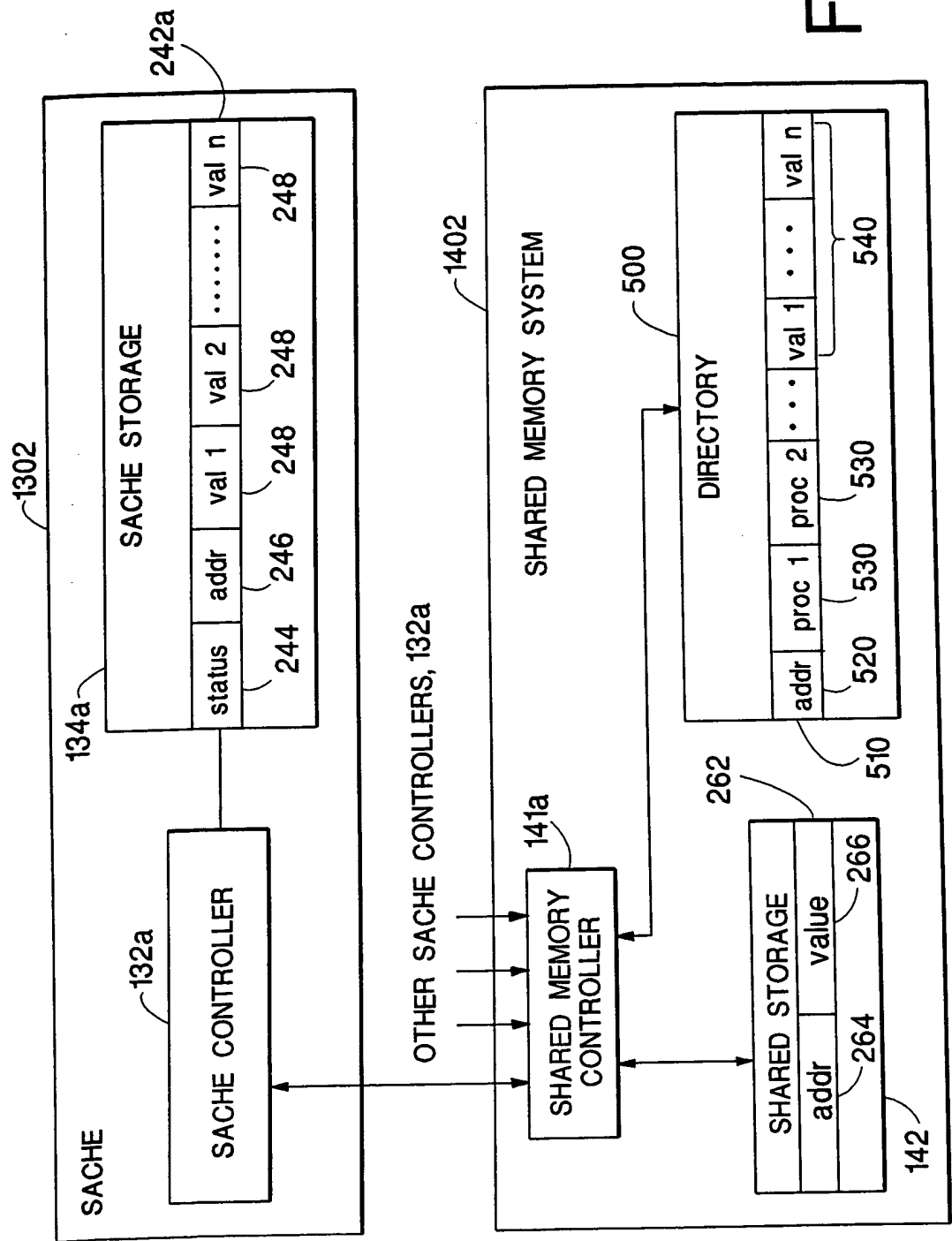


FIG. 5

Process LoadL(addr): **610**

```
611 IF Cell(addr,-,Invalid) THEN
612     CREATE Cell(addr,-,Cache-Pending)
613     SEND Cache-Request(addr) message
614     STALL LoadL processing UNTIL Cache message is processed
615     GET val from Cell(addr,val,Clean)
616     RETURN val
617 ELSE IF Cell(addr,val,Clean) OR Cell(addr,val,Dirty) THEN
618     RETURN val
619 ENDIF
```

FIG. 6A

Process Reconcile(addr) message: **630**

```
631 RETURN ack
```

FIG. 6B

Process StoreL(addr, val) message: 640

```
641  IF Cell(addr, -, Invalid) THEN
642      CREATE Cell(addr, -, Cache-Pending)
643      SEND Cache-Request(addr) message
644      STALL StoreL(addr) processing UNTIL Cache(addr) message
645  END IF
646  SET Cell(addr, val, Dirty)
647  RETURN ack
```

FIG. 6C**Process Commit(addr) message: 650**

```
651  IF Cell(addr, val, Dirty) THEN
652      SET Cell(addr, val, Writeback-Pending)
653      SEND Writeback(addr, val) message
654      STALL Commit(addr) processing UNTIL Writeback ack
655      SET Cell(addr, val, Clean)
656  END IF
657  RETURN ack
```

FIG. 6D

Process Cache(addr, val): 670

```
671     IF Cell(addr, -, Invalid) THEN
672         CREATE Cell(addr, -, -)
673         SET Cell(addr, val, Clean)
674     ELSE IF Cell(addr, -, Cache-Pending)
675         SET Cell(addr, val, Clean)
676         RESTART STALLED LoadL(addr) and StoreL(addr)
677     ENDIF
```

FIG. 6E

Process Writeback-Ack(addr) message: 680

```
681     IF Cell(addr, val, Writeback-Pending) THEN
682         SET Cell(addr, val, Clean)
683         RESTART STALLED Commit(addr) processing
684     END IF
```

FIG. 6F

Process Purge-Request(*addr*) message: 690

```
691   IF Cell(addr,-,Clean) THEN
692       DELETE addr from sache storage
693       SEND Purged(addr) message
694   ELSE IF Cell(addr,-,Dirty) THEN
695       SEND Writeback(addr) message
696       DELETE addr from sache storage
697   ENDIF
```

FIG. 6G